

Results from a Program for Spectral
Analysis using a Graphic Display

Bertil Ekenberg

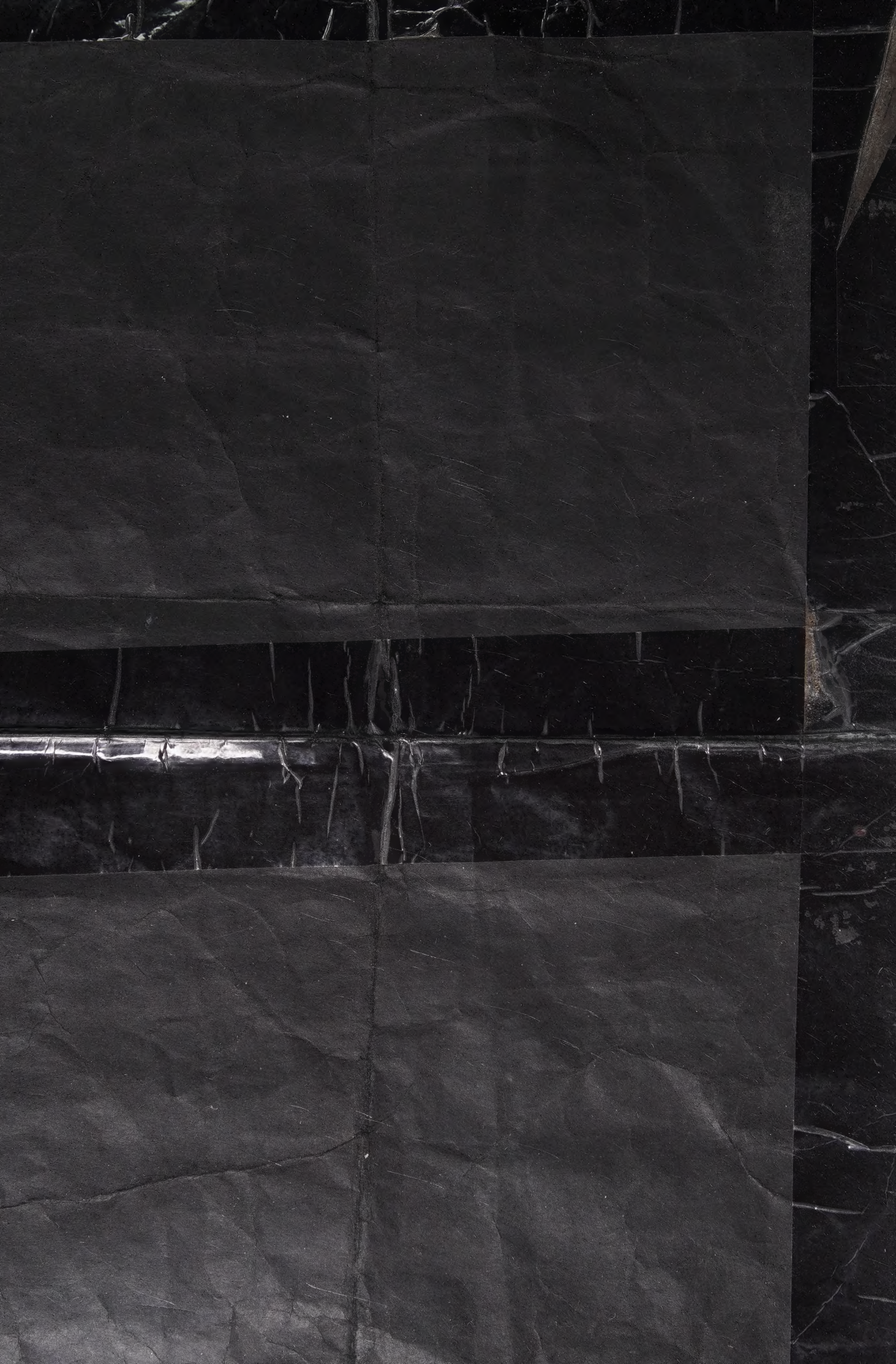
Report LU/CS 75-5

May 1975



LUND UNIVERSITY
Department of Computer Sciences

1978



Lund University

Department of Computer Sciences

Sölvegatan 14A

S-223 62 Lund

Sweden

Results from a Program for Spectral
Analysis using a Graphic Display

Bertil Ekenberg

Report LU/CS 75-5

May 1975

Lund University
Department of Computer Sciences
Sölvegatan 141
S-223 62 Lund
Sweden

Research from a Project for the
Analysis using a Graphical Language

Per Erik Hansson

Report 11/83-10-3

May 1983



Abstract

This report presents results from a program for spectral analysis using a graphic display unit. In spectral analysis measurements may be made and may be presented as a curve. However, in this report the curve is constructed by another program in order to simplify comparisons between the constructed curve and the results from the program for spectral analysis. The curve is regarded as a sum of Gaussian curves, and the problem is to determine parameters of these curves. The curve is displayed on the display screen, and the values of parameters may be suggested by the person at the display console. Different optimization methods are used to improve the fit. Then the original curve and the latest calculated curve are presented on the display screen.

For one example discussed in this report it was found that although the curves were very close to each other, the parameters of the curves were far from each other. If the original curve would not have been calculated, but obtained from measurements, the errors of the measurements would probably have been much greater than the errors of the fit, and it would probably be impossible to conclude that the parameters were almost correct even if the best fit was found.

Contents

Introduction and summary	1
The spectral program	2
The test generator	3
Results from one curve fitting	4
Results from other curve fittings	9
Conclusions	12
Questions open for discussion	13
Appendix 1: Commands for the spectral program	14
Appendix 2: Description of some numerical methods used in the spectral program	16
References	21

Introduction and summary

Measurements from spectral analysis may be presented as a curve, for example

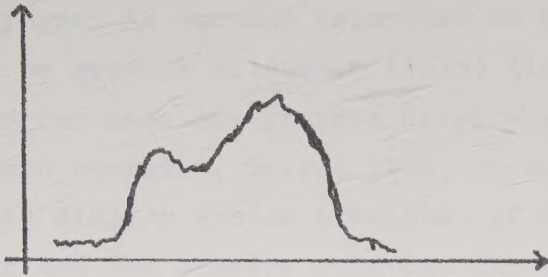


fig. 1

The horizontal axis corresponds to different energy levels and the vertical axis to the intensity on different levels. If disturbances are disregarded, the curve is supposed to be a sum of Gaussian curves, i.e. a function of the form

$$f(t) = \sum_{k=1}^n a_k e^{-c \cdot (\beta_k - t)^2 / \gamma_k^2} \quad (1)$$

or a sum of Lorentz curves, i.e. a function of the form

$$f(t) = \sum_{k=1}^n \frac{a_k}{1 + \gamma_k (\beta_k - t)^2 + \delta_k (\beta_k - t)^4 + \varepsilon_k (\beta_k - t)^6} \quad (2)$$

and possibly a straight line (or different straight lines in different parts of the curve). The problem is to determine the number of curves (n) and the parameters a_k , β_k , and γ_k in (1) and a_k , β_k , γ_k , δ_k , and ε_k in (2) and possibly the parameters of the straight line (or lines).

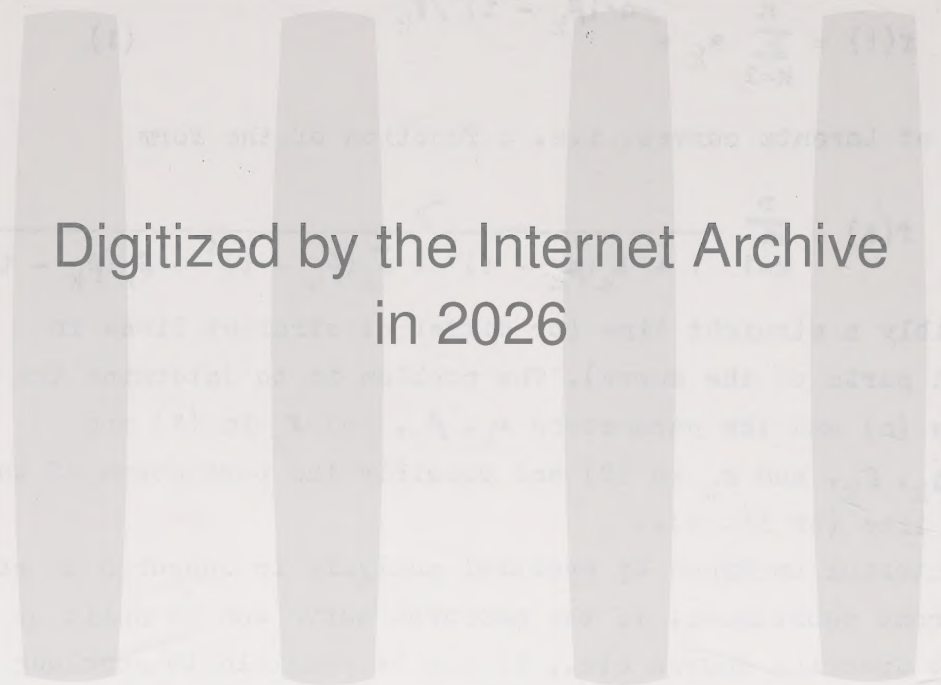
The material analysed by spectral analysis is supposed to consist of different substances. If the measured curve can be split up in different Gaussian curves etc., it may be possible to conclude what substances are present.

A computer program has been written in order to simplify the analysis. When the program was tested, a constructed curve (only two Gaussian curves without disturbances) was used instead of a measured curve, and efforts were made to find the parameters of the constructed curve. In such a test it was found that this curve and a similar curve were very close to each other although the parameters were far from each other. The test is described in this report. The conclusion is that it is very difficult to determine the parameters even when the curves are not disturbed, and it seems to be impossible if the values are disturbed so that the errors in the measurements are greater than the errors in the approximation. In this case some extra knowledge of the parameters must be available, probably in the form of constraints.

Introduction and Summary
Measurements from spectral analysis may be presented as a curve,
for example



FIG. 1
The horizontal axis corresponds to different energy levels and the
vertical axis to the intensity of radiation. The curve is
distorted and the intensity of the curve is supposed to be a sum of
several curves, for example, $I = I_1 + I_2 + \dots$



Digitized by the Internet Archive
in 2026

The spectral program

This program is further described in Ekenberg: Spektralanalys med hjälp av grafisk bildskärm (1975) (in Swedish).

The program uses the graphic display system, Univac 1557/1558, and the main computer, Univac 1108, at Lund Computing Center, Lund. The graphic display system consists of two parts, the display controller and the display console. The display controller may be regarded as a minicomputer. It has a memory (16 K 18-bits-words, cycle time $0.7 \mu s$) in which picture descriptions and part of the spectral program is stored. The picture is drawn on the display screen as lines and characters. The characters are both ordinary characters, such as letters, digits, and parentheses, and special characters, such as some Greek letters, integral sign, root sign and so on. The corresponding characters may be written on the display keyboard, which looks like an extended typewriter keyboard. The program in the display unit examines the messages written on the keyboard, produces the pictures to be displayed, and makes simple calculations. If more elaborated calculations are required, necessary data are sent to the program in the main processor where the computations are made, and the result is sent back to the display unit.

When the program starts, it reads the values of the measurements and presents the corresponding curve at the display screen. Then the user may guess the parameters of the curve and the function form, send the parameters to the main computer, which returns the curve corresponding to these parameters. After that it is possible to continue guessing or to let the computer guess curves or optimize the curve fitting by different numerical methods. There is also a possibility to change parameters of the optimization routines, to change the amount of output on the line printer and so on. When the approximation is completed, the user writes a command which stops the program.

The test generator

When the program was tested it was convenient to use a constructed curve instead of a measured curve as input curve. In this way the result of the fit could be compared to the constructed curve, the form of the curve and the errors of the curve values could be controlled and the approximate size of some parameters was known beforehand. However, the person at the display console should not know the parameters of the constructed curve exactly since this knowledge might influence the guessing of parameters and choice of methods too much.

Therefore a program generating test data was written. The number of curves, the values of the parameters t , β , γ and, if necessary, J and S and limits for a straight line, and the values of the disturbances for each function value was chosen at random within intervals read from card reader. Further, the function form (form (1) or (2), with or without a straight line) could be chosen at random. The values of the curve were stored on a disk memory and then read by the spectral program, and the actual parameters were written on a line printer but not read by the person at the console when the test started.

Results from one curve fitting

A curve fitting with a typical negative result when two peaks are relatively close to each other is described.

The test generator just mentioned was used with parameters chosen to give a curve of form (1) with $n=2$, $50 \leq \alpha \leq 100$, $1 \leq \beta \leq 4$, $0.5 \leq \gamma \leq 2$, without disturbances. The number of values of the curve (the number of "measure points") should be between 50 and 100. In a curve of the form

$$f(t) = \alpha e^{-c \cdot (\beta - t)^2 / \gamma^2} \quad (3)$$

α corresponds to the height of the peak, β to the location of the peak and γ to the deviation of the peak. If c is chosen to be $4 \cdot \ln(2)$, as in this case, γ is equal to the width of the peak when $f(t) = \alpha/2$

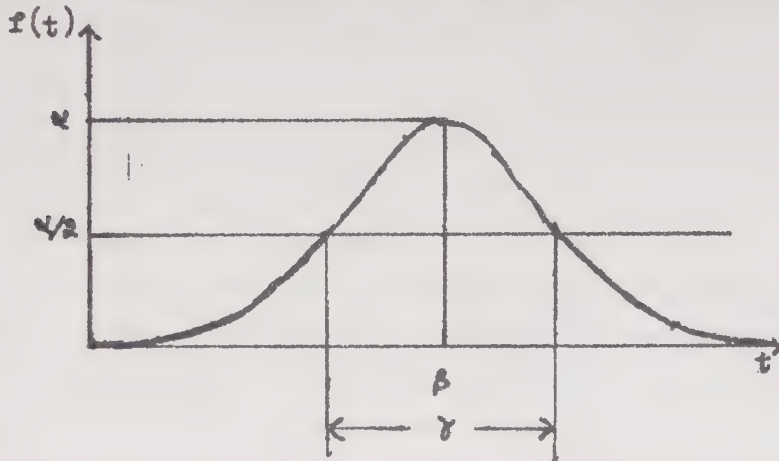


fig. 2.

The test generator chose the following parameters: $\alpha_1=89.13259$, $\beta_1=3.41095$, $\gamma_1=1.66706$, $\alpha_2=71.76958$, $\beta_2=2.82991$ and $\gamma_2=1.61194$. The number of values of the curve was 55, and the step length between two values was 0.1 starting at 0, so that the curve was calculated for values 0.0, 0.1, 0.2, ..., 5.4.

When the program started, the curve in fig. 3 was displayed on the display screen.

Although the curve consists of two Gaussian curves, it looks like one Gaussian curve. So an effort was made to fit the curve to one Gaussian curve, and after some calculations the result was $\alpha_1=147.9355$,

$\beta_1=3.1527$, $\gamma_1=1.7900$, the error measured as $\sum_{i=1}^n (y_1(i) - y_2(i))^2 / n$ was 0.1143, the error $\sqrt{\sum_{i=1}^n (y_1(i) - y_2(i))^2}$ was 2.5073 and the greatest absolute error $\max_{1 \leq i \leq n} |y_1(i) - y_2(i)|$ was 0.6900, where $y_1(i)$ are the values of the original curve and $y_2(i)$ the values of the

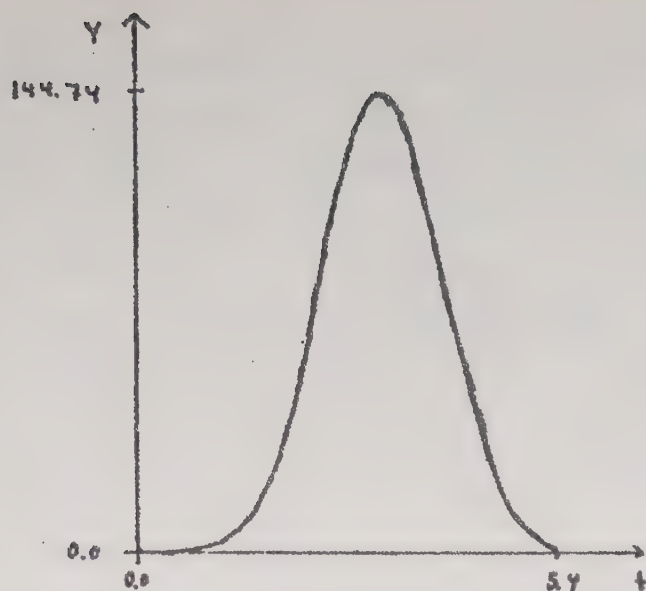


fig. 3.

approximating curve. (The first form of the error is henceforth called E_1 , the second form E_2 , and the third form E_3 .) The curves come very close to each other, and nothing was found in the results that would indicate that the curve consists of two Gaussian curves. The errors are probably much smaller than the errors from real measurements. If it was not known that the curve should be fitted to two Gaussian curves, this result could very well be accepted as a final value.

This curve was roughly divided into two curves and the new parameters were sent to the main computer and an approximation was made by Nedler-Mead's Simplex method (1965). This method and some other methods used by the program are described in appendix 2. The computer returned the values $\mu_1=81.1020$, $\sigma_1=3.2111$, $\tau_1=1.7813$, $\mu_2=67.5251$, $\sigma_2=3.0817$, $\tau_2=1.7795$, and the errors $E_1=0.1199$, $E_2=2.5680$ and $E_3=0.6288$. The difference curve is still very small although the parameters differ considerably from the parameters chosen by the test generator. The errors E_1 and E_2 are slightly worse than before because the person at the display console sent bad parameters as a start approximations to the Simplex method. A test routine, Gausstest, was called to find out if the error E_1 (or E_2) was close to a minimum, but the test failed. The Simplex routine was called again, but the changes of the parameters were less than 0.1 percent. The error E_1 became 0.1181, E_2 2.5487, and E_3 0.6427. Gausstest failed again, and another call on Simplex did not change the values very much (E_1 became 0.1168, E_2 2.5346, E_3 0.6545, all parameters changed < 0.1 percent). The parameter EPS, which determines the accuracy of the Simplex method (maximum error-minimum error of the last examined points), was changed from 0.01 to 0.0001 from the display console, and the Simplex method was used again.

This time the parameters were changed to $\alpha_1=80.8118$, $\beta_1=3.2160$, $\delta_1=1.8256$, $\alpha_2=67.7446$, $\beta_2=3.0793$, $\delta_2=1.7310$, the error $E_1=0.07659$, $E_2=2.0524$, $E_3=0.5088$. Although the errors are reduced, the parameters are still far from the values chosen by the test generator.

The α_1 values differ most (80.2 instead of 89.1), and δ_1 changed from 1.78 to 1.82 while the original value should be 1.67. However, the area under a peak is $\int_{-\infty}^{\infty} \alpha e^{-c(\beta-t)^2/\delta^2} dt = \text{constant} \cdot \alpha \cdot \delta$, and $\alpha_1 \cdot \delta_1$ is 147.53 for the last curve and 148.59 for the curve from the test generator (the corresponding values for $\alpha_2 \cdot \delta_2$ are 117.31 and 115.69). The δ_2 value was changed in the correct direction, and the other values were not changed very much. The peaks are too close to each other. The difference between the β values is 0.1367 instead of 0.5810. This error probably influences the α and δ values very much.

In this situation another method was tried several times. The program called the subroutine Gaussa, which is written and described by Wedin-Carlsson (1972). The routine uses the Gauss-Newton method, described by Wedin (1972). Every time the computations were stopped by the error condition maximum time used for the routine, and the last computed value was sent back. The maximum time in this case was approximately 1 second CPU-time. The routine searched all the time in the wrong directions and increased α_1 and decrease α_2 so that the error E_2 decreased. The β and δ values remained almost unchanged. The results from the computations were:

α_1	β_1	δ_1	α_2	β_2	δ_2	E_1	E_2	E_3
94.0153	3.2125	1.8127	54.7241	3.0522	1.7241	0.07118	1.9786	0.4956
104.5125	3.2075	1.8057	44.3603	3.0258	1.7143	0.06754	1.9273	0.4855
111.3817	3.2065	1.7998	37.6918	2.9966	1.7057	0.06349	1.8687	0.4637
119.7936	3.2068	1.7907	29.7303	2.9391	1.6895	0.05794	1.7852	0.4222

Since the α values should be between 50 and 100, an effort was made to find out if other methods would search in other directions.

Nedler-Mead's Simplex method was tried twice, at first with $\text{EPS}=0.01$ and then with $\text{EPS}=0.00001$, and a routine Stepit by Chandler (1965) was tried once, but the changes of the parameters were only about 1 percent. The Gauss-Newton method was tried twice more but continued as before. The results from these five optimizations were:

α_1	β_1	δ_1	α_2	β_2	δ_2	E_1	E_2	E_3
119.7891	3.2068	1.7910	29.7307	2.9389	1.6895	0.05758	1.7796	0.4331
120.3835	3.2150	1.7909	29.7297	2.9036	1.6516	0.04484	1.5704	0.3097
120.3921	3.2148	1.7907	29.7376	2.9039	1.6523	0.04443	1.5631	0.3037
126.3128	3.2098	1.7840	24.1409	2.8523	1.6290	0.03694	1.4254	0.3076
132.4030	3.2016	1.7793	18.2714	2.7911	1.5926	0.03094	1.3045	0.3485

The difference between the β values has increased, and the γ values are getting closer to the produced values, but the α values are very far from the correct values. The Simplex method was called again with EPS=0.00001, and the result was $\alpha_1=133.1470$, $\beta_1=3.2061$, $\gamma_1=1.7766$, $\alpha_2=18.0823$, $\beta_2=2.7499$, $\gamma_2=1.5483$, $E_1=0.02111$, $E_2=1.0775$, and $E_3=0.2949$.

An effort was made to optimize with fewer function values. Only the 19 function values $f(t)$ for $t=2.2, 2.3, \dots, 4.0$ were used when the Gauss-Newton method was called. The parameters changed to $\alpha_1=133.4397$, $\beta_1=3.2210$, $\gamma_1=1.7582$, $\alpha_2=19.7707$, $\beta_2=2.6618$, $\gamma_2=1.5169$, and the errors for these 19 function values $E_1=0.001401$, $E_2=0.1631$, $E_3=0.0598$ and the error for all the 55 functions values changed to $E_1=0.006321$, $E_2=0.5896$, and $E_3=0.0621$. After that, all function values were used again, Gausstest was made, but failed, and the Gauss-Newton method was used twice, and then the Simplex method was used (with EPS=0.01). The parameters and errors became respectively

α_1	β_1	δ_1	α_2	β_2	γ_2	E_1	E_2	E_3
126.3336	3.2508	1.7409	28.5916	2.6941	1.5547	0.002173	0.3457	0.1041
121.1657	3.2720	1.7306	34.8086	2.7161	1.5704	0.001612	0.2977	0.0852
121.1651	3.2720	1.7309	34.8081	2.7161	1.5707	0.001700	0.3058	0.0926

This curve is very close to the curve generated by the test generator, but the α values are still outside the permitted interval $50 \leq \alpha \leq 100$. Although the α values changed in the right direction, the person at the console decided to "help" the program to find the produced values and looked at the parameters that the test generator had written. As mentioned before, the values were $\alpha_1=89.13259$, $\beta_1=3.41095$, $\gamma_1=1.66706$, $\alpha_2=71.76958$, $\beta_2=2.82991$ and $\gamma_2=1.61194$. α_1 was changed to 89 and α_2 to 71, and this curve was presented at the display screen. It was obvious that the β values were too small, and Gausstest was made, but failed again, and the Simplex routine was called. The parameters became $\alpha_1=90.9205$, $\beta_1=3.3595$, $\delta_1=1.7201$, $\alpha_2=75.0439$, $\beta_2=2.8673$, $\gamma_2=1.6409$, and the errors changed from $E_1=78.58$, $E_2=65.74$, $E_3=16.41$ to $E_1=0.03430$, $E_2=1.3735$, $E_3=0.4442$. Since the message that Simplex used the maximum number of iterations was sent to the display screen, the routine was called again, but the parameters were changed less than 0.1 percent. The Gauss-Newton method was used and gave the result $\alpha_1=89.1364$, $\beta_1=3.4109$, $\delta_1=1.6671$, $\alpha_2=71.7656$, $\beta_2=2.8299$, $\gamma_2=1.6119$, which is almost exactly the values that the test generator produced. The errors were $E_1=0.00000007$, $E_2=0.00197$ and $E_3=0.0006$. This time the method reached sufficient accuracy, and no more optimizations were made.

As mentioned before, the Gauss-Newton method started at last to search in the right direction, but the person at the console did not notice that until after the run and gave better parameter values instead. After the run an effort was made to continue with the old parameter values in order to find if and when the best parameter values were found. The Gauss-Newton method was tried several times, and the results were as follows (the first line is the old start values).

α_1	β_1	γ_1	α_2	β_2	γ_2	E_1	E_2	E_3
121.1651	3.2720	1.7309	34.8081	2.7161	1.5707	0.001700	0.3058	0.0926
115.8153	3.2942	1.7199	41.1807	2.7373	1.5820	0.001262	0.2634	0.0687
112.2645	3.3091	1.7132	45.3674	2.7503	1.5876	0.000929	0.2261	0.0619
104.1768	3.3438	1.6970	54.7866	2.7791	1.5978	0.000470	0.1607	0.0505
90.7578	3.4034	1.6704	69.9748	2.8246	1.6106	0.000083	0.0676	0.0209
89.1346	3.4109	1.6671	71.7676	2.8299	1.6119	0.000000	0.0004	0.0001

The last time the Gauss-Newton method stopped when sufficient accuracy was obtained. The error E_1 in this case was $3.1 \cdot 10^{-9}$.

The parameter values are now almost exactly the same as those produced by the test generator. The Gauss-Newton method was tried twice more, with the accuracy (ACCU) changed to 10^{-6} and 10^{-10} and in both cases the method stopped when the rounding errors were greater than the changes of the parameters, and in both cases the parameters and errors were changed to $\alpha_1=89.1334$, $\beta_1=3.4110$, $\alpha_2=71.7659$, $E_1=1.09 \cdot 10^{-9}$, $E_2=2.45 \cdot 10^{-4}$, and $E_3=8.01 \cdot 10^{-5}$ and the other values were not changed. A final call for the Simplex method did not improve the result.

Results from other curve fittings

This example shows how the program works when the peaks are separated from each other.

The test generator was used again with the same input parameters as before, and this time the curve with the parameter values $\mu_1=76.66948$, $A_1=2.52642$, $\gamma_1=0.87850$, $\mu_2=65.97176$, $A_2=3.97588$, $\gamma_2=0.61526$, and the number of points=57, was produced. The curve is presented in fig. 4.

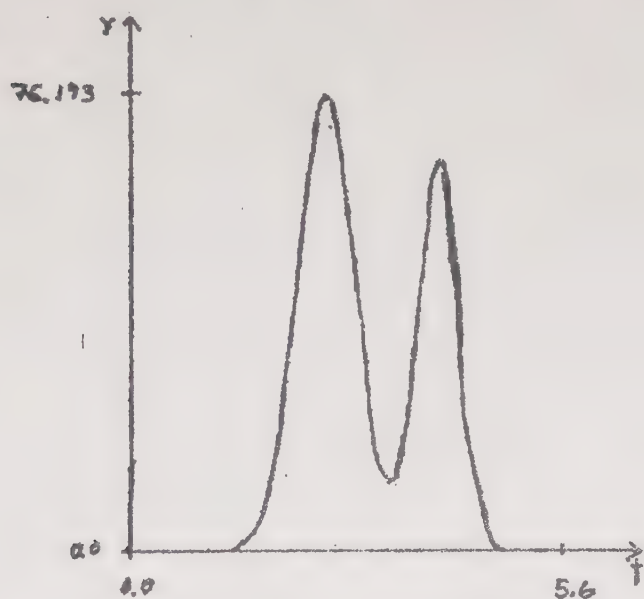


fig. 4.

At first some variables, for example those determining the printing of the results, were changed from the console, and then the program was asked to guess two peaks. After that the parameter value and errors were $\mu_1=76.4775$, $A_1=2.5288$, $\gamma_1=0.9178$, $\mu_2=65.6610$, $A_2=3.9808$, $\gamma_2=0.6180$, $E_1=0.9631$, $E_2=7.4091$, and $E_3=2.6087$. The Gauss-Newton method was used, and the parameter values became the same as those produced by the test generator with an absolute or relative error $< 10^{-4}$. The errors became $E_1=4.639 \cdot 10^{-10}$, $E_2=1.626 \cdot 10^{-4}$, and $E_3=6.314 \cdot 10^{-3}$. This shows that when the peaks are separated, the program easily find the correct parameter values.

In another example the produced curve seemed to be just as difficult to analyse as the first curve of this report, but the program soon found the correct values.

The test generator used the same input parameters as before and produced the curve with the parameter values $\alpha_1=57.53861$, $\beta_1=2.50158$, $\gamma_1=1.46932$, $\alpha_2=68.62627$, $\beta_2=2.25775$, $\gamma_2=0.74416$ and the number of points=71. The curve is drawn in fig. 5.

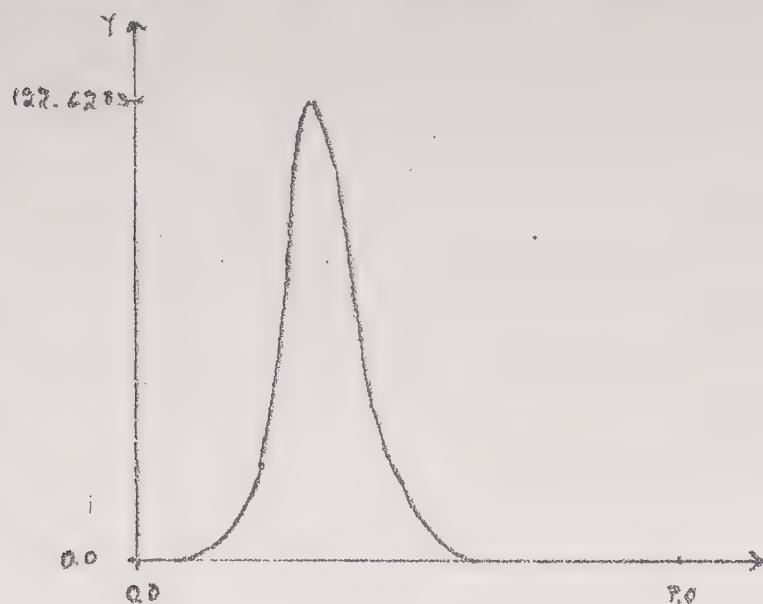


fig. 5.

The program was asked to guess one peak, and the values of the peak became $\alpha_1=122.6283$, $\beta_1=2.3455$, and $\gamma_1=1.0613$. When the result was presented at the display screen, it was seen at once that the fitting was bad, and continued to be bad even when the Gauss-Newton method was tried. When the parameters were $\alpha_1=117.0073$, $\beta_1=2.3491$, and $\gamma_1=1.1068$, the person at the console guessed parameter values and changed the value of α_1 and asked the computer to draw the curve corresponding to the values $\alpha_1=60$, $\beta_1=2.3491$, $\gamma_1=1.1068$, $\alpha_2=60$, $\beta_2=2.5$, and $\gamma_2=1$. The α values were chosen to 60 since the maximum function value was 122.6283 and the α values should be between 50 and 75, and β_2 was chosen somewhat greater than β_1 , since the difference between the curves was greatest at the right of the curve peak.

The errors were still great. With E_1 , E_2 , and E_3 defined as before, the errors were $E_1=39.4527$, $E_2=52.9258$, and $E_3=16.7206$. The Gauss-Newton method was tried, and gave the result $\alpha_1=75.9143$, $\beta_1=2.3874$, $\gamma_1=1.2470$, $\alpha_2=59.3465$, $\beta_2=2.2775$, and $\gamma_2=0.9236$ when the maximum time for the routine was used. The errors were still rather great ($E_1=11.1719$, $E_2=28.1638$, $E_3=8.8591$), and the Gauss-Newton method was used again. This time sufficient accuracy was obtained, and the result was $\alpha_1=57.5387$, $\beta_1=2.5016$, $\gamma_1=1.4693$, $\alpha_2=68.6264$, $\beta_2=2.2578$,

and $\chi_2=0.7442$, and the errors $E_1=1.3876 \cdot 10^{-9}$, $E_2=3.1386 \cdot 10^{-4}$, and $E_3=1.2207 \cdot 10^{-4}$. It was easily seen at the display screen that the fitting was successful. The Gauss-Newton method was used again with the variable $ACCU=10^{-6}$, but the routine only changed χ_2 to 68.6263 and the errors to $E_1=2.0109 \cdot 10^{-10}$, $E_2=1.1949 \cdot 10^{-4}$, and $E_3=4.5776 \cdot 10^{-5}$ and stopped when the rounding errors were as small as the changes of the parameters.

Conclusions

This report demonstrates the difficulties of determining the parameters height, location and deviation of peaks close to each other when the peak curves are added to each other. It was known that a curve consisted of two peaks, and the function form describing the peaks was known. Nevertheless, many parameter combinations were found far from the correct parameters and far from each other, but the differences between these curves and the original curve were very small. In this report the original curve was constructed to be exactly a sum of two Gaussian curves, but when measurements are made in spectral analysis, the values of the curve are disturbed and the function form is not known exactly and is only approximately Gaussian curves. These errors are probably much greater than the errors due to wrong parameters as above. In this case it seems to be impossible to conclude that the correct parameters are found, even if the parameters with minimum difference between the curves are found. It seems to be necessary to know more about the curves to get approximately correct parameter values.

Questions open for discussion

Now the problem is to determine what kind of knowledge that is needed and how the knowledge should be included in the program. One possibility is to add constraints determining connections between the parameters. Connections between two or more peaks may be caused by physical or chemical properties of the examined material. For example, since the area under a peak is proportional to $\kappa \cdot \gamma$, the relation between the areas of two peaks may be expressed as an identity as $\kappa_1 \cdot \gamma_1 = \kappa_2 \cdot \gamma_2 \cdot 0.9$ or inequalities as $\kappa_1 \cdot \gamma_1 - \kappa_2 \cdot \gamma_2 \cdot 0.9 < 0.1$ and $\kappa_1 \cdot \gamma_1 - \kappa_2 \cdot \gamma_2 \cdot 0.9 > -0.1$. The distance between two peaks may be described as $\beta_1 - \beta_2 = 0.4$ or $\beta_1 - \beta_2 > 0.35$.

There may also be another kind of knowledge, for example that it seems reasonable that the height of a peak is less than 100, which may be expressed as $\kappa_1 < 100$, or that the top of a peak is between 3.5 and 3.9, which may be written $\beta_2 > 3.5$ and $\beta_2 < 3.9$.

How should this knowledge be included in the program? If identities are used, must they hold exactly in all calculations, or may they have errors of the same size as the function values? How many constraints may be provided? Are all of them equally important? If not, the user may have to include the conditions in the least squares problem as $100 \cdot (\beta_1 - \beta_2 + 0.4) = 0$ if the error in the condition $\beta_1 - \beta_2 + 0.4 = 0$ should be approximately 100 times smaller than the mean error in the function values. Perhaps a penalty function, as $100 \cdot (\beta_1 - \beta_2 + 0.4)^2 + 10 \cdot (\kappa_1 - 75)^2 + 0.1 \cdot (\gamma_1 - 1.1 \cdot \gamma_2)^2$, should be added to the error of the function values.

There may also be a possibility to find the best fitting point or points by calculating the function values of many points within an area. The person at the console may determine for example what parameters to vary and their start and end values and step lengths. However, there are some problems. What results should be presented and how should they be presented to the user so that the user may get an opinion of where to search further for minimum points?

Appendix 1

Commands for the spectral program

The user at the display unit decides what the computer shall do by writing commands on the display console keyboard. Some of the commands are listed below (translated from Swedish into English). The complete list is found in Ekenberg: Spektralanalys med hjälp av bildskärm (1975). Many commands may be written in a short form, since the program may recognize a command when for example the first three characters are written.

variable

This command means a question of the value of a variable. The value is written on the display screen.

variable=number

The variable is given the value of the number, which may be written as a number in Fortran or Algol. The variable in this and the previous command may determine the amount of printing on the line printer, accuracy, step lengths and other constants of the numerical methods, function form (Gaussian or Lorentz curves, with or without a straight line), function values of the original curve and the approximating curve, the values of the α , β , γ , δ , and ϵ parameters, and so on.

PICTURE number version

The pictures, consisting of the original curve, the approximating curve, the difference between the curves, the function form and the α , β , γ , δ , and ϵ values are saved in the display unit when the curves or the parameters are changed, and the latest pictures may be recalled to the display screen.

DIFFERENCE

means that the difference curve is to be taken as the original curve henceforth. In this way a background curve or some of the peaks may be removed after fitting, while the user and the program continues fitting the other peaks. If the curve with the first measured values is to be displayed again, the command

OLD

is written. Different optimization routines are called by

GAUSS

SIMPLEX

and

STEPIT

The first of these commands uses the Gauss-Newton method, the second Nelder-Mead's Simplex routine and the third Chandler's routine Stepit. The routines are described in appendix 2. It is possible to let the program choose between the Gauss-Newton method and Simplex by writing the command

GAUSSTEST

or

TEST

or to perform the test and send the result to the display unit. Some of the parameters may be fixed, that is they are not changed by any optimization routine, by the command

FIX

followed by the parameters to be fixed. If the parameters should be permitted to change again, the parameters should be written after the word

NOTFIX

There is also a possibility to let the program guess the parameter of a new peak and add the peak to the old approximating curve by the command

GUESS

One peak is added each time the command is used. If the parameters are changed by the user and the corresponding curve is to be calculated and displayed at the screen, the user writes

NEW CURVE

When one curve is fitted sufficiently well, and a curve from another measurement should be fitted, these curve values are read from a card reader or disk file by the command

NEW READING

and when the last curve is fitted the program is stopped after the command

STOP

and the user gets the results of the program at the line printer.

Appendix 2

Description of some numerical methods used in the spectral program

The Gauss-Newton method

This method is described by Wedin-Carlsson (1972).

A solution to a non-linear least squares problem is solved by using Gauss method repeatedly. In each step the problem is approximated to a linear least squares problem and Amijo's step length principle is used.

The method may also be described as follows: Let $y_1(1), y_1(2), \dots, y_1(n)$ be the measured values for $t=t_1, t_2, \dots, t_n$, and $y_2(i) = \sum_{k=1}^r u_k e^{-c \cdot (\beta_k - t_1)^2 / \delta_k^2}$ approximate $y_1(i)$. Each function $f_1 = y_1(i) - y_2(i)$ is a function of $3 \cdot r (=m)$ variables $u_1, \beta_1, \delta_1, \dots, u_r, \beta_r, \delta_r$, and $f = (f_1, f_2, \dots, f_n)^T$ is thus a function from R^m to R^n . In this case $\phi = \sum_{i=1}^n f_i^2$ is to be minimized, that is a combination of $u_1, \dots, \delta_r$ is to be found so that the vector $(f_1, f_2, \dots, f_n)$ is as close to the origin $(0, 0, \dots, 0)$ in the n -dimensional space as possible.

Let x denote a vector $(u_1, \dots, \delta_r)$ in the m -dimensional space, and let A denote a $n \times m$ -matrix with $a_{ik} = \frac{f_i}{x_k}$, and let h denote a vector in the m -dimensional space. Then $f(x+h)$ is approximately equal to $f(x) + Ah$ if h is close to the origin, and the tangent plane through a point x_0 may be described as the vectors $v = f(x_0) + A(x_0)h$ for arbitrary h in R^m . The vector from $f(x_0)$ to the origin in R^n is projected onto this tangent plane, and the projected vector (in R^m) is used to change the old vector x_0 .

If the situation is simplified to the case $m=2, n=3$ (that is 2 variables, say u_1 and β_1 , may be changed in 3 points f_1, f_2, f_3 corresponding to 3 different values of t, t_1, t_2 and t_3), it may be illustrated by a figure:

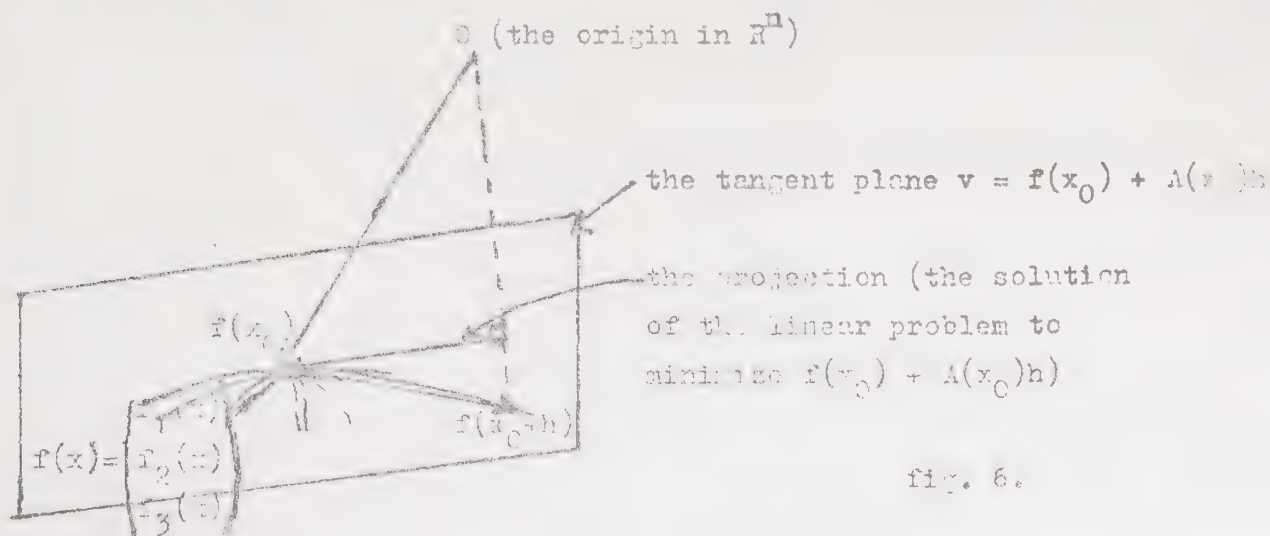


fig. 6.

If the projection of the vector from the origin in $\mathbb{R}^n$ to $f(x_0)$ is denoted $P_{R(A)}f(x_0)$, the point (in $\mathbb{R}^n$) in the tangent plane closest to the origin becomes $f(x_0) - P_{R(A)}f(x_0)$ (which is computed as $f(x_0) - A A^+ f(x_0)$, where A^+ is the pseudo-invers of A ($A^+ = (A^T A)^{-1} A^T$)). The corresponding vector x (in $\mathbb{R}^n$) is changed in the Gauss direction $-A^+ f(x_0)$.

Sometimes the gradient direction is used (x is changed in the direction $-A^T f(x_0)$) instead of the Gauss direction so that convergence towards a point where A is singular is avoided. Another reason is explained by an example. Suppose that f is a function from $\mathbb{R}$ to $\mathbb{R}^2$ that only takes values on a circle (with the origin denoted O in the figure). The point A on the circle closest to the origin O in $\mathbb{R}^2$ is to be found (A is the point in $\mathbb{R}^2$ that minimizes $\|f(x)\|_2^2 = \sum f_i^2(x)$). If P is the start point, the point R is the point on the tangent through P that is closest to O , and S is the next start point if the method takes a step in the Gauss direction. But if the start point is T , the point on the tangent through T is T , and the Gauss direction can not be used in this case, so the gradient direction is used instead. If the start point is close to T and the Gauss direction is used, only small improvements are achieved.

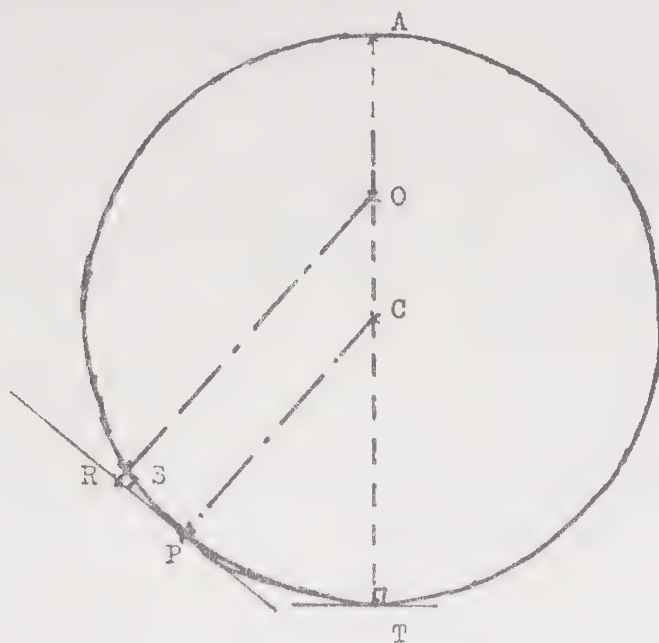


fig. 7

(In the first curve fitting problem in this report the Gauss-Newton method converged very slowly. Probably the method used the Gauss direction but the start point was similar to a start point near T in this example, but not close enough to T to force the method to use the gradient direction.)

The method continues to take Gauss steps or gradient steps until the error of the fitting is smaller than a constant (ACCU) or the maximum time for the method has been exceeded.

Gausstest

Let the functions and the variables be defined as in the description of the Gauss-Newton method. Let $G^{(j)}$ denote the second derivatives of f_j , so that $G_{ik}^{(j)} = \frac{\partial^2 f_j}{\partial x_i \partial x_k}$. The matrix $H = A^T A + \sum_{j=1}^n ((I - P_{R(A)}) f)_j G^{(j)}$ is computed. If H is positive definite, the test is said to succeed, otherwise the test is said to fail. If the command TEST or GAUSSTEST is written on the console keyboard, the test is made, and if the test succeeds the Gauss-Newton method is used, otherwise the Simplex method is chosen.

Simplex method

Simplex method in this report means the method described by Nelder-Mead in A Simplex method for Function Minimization (1965). The method is somewhat changed, mainly according to Parkinson and Hutchinson (1971).

Let $y_1(1), y_1(2), \dots, y_1(n)$ be the measured values for $t=t_1, t_2, \dots, t_n$, and let $y_2(i) = \sum_{k=1}^r \alpha_k e^{-c \cdot (\beta_k - t_i)^2 / \delta_k^2}$ approximate $y_1(i)$. $f = (\sum_{i=1}^n (y_1(i) - y_2(i))^2) / n$ is to be minimized as a function of $3 \cdot r (=m)$ variables $\alpha_1, \beta_1, \delta_1, \dots, \alpha_r, \beta_r, \delta_r$.

At first the values $f_0, f_1, \dots, f_m$ in the start point and n surrounding points $P_0, P_1, \dots, P_m$ are calculated. The main idea is that a number of steps are taken and in each step the point with the greatest value ($f_{\max}$) is replaced by a point with a smaller function value.

At first $f_{\max}$ (with the corresponding point $P_{\max}$) and $f_{\min}$ ($P_{\min}$) are found and $\bar{P} = (\text{the mean value of } P_i \text{ except } P_{\max}) = \sum_{k \neq \max} P_k / (m-1)$ is calculated. Then $P_r = (1+A) \cdot \bar{P} - A \cdot P_{\max}$ and the function value f_r are calculated. If $m=2$ the situation may be illustrated by fig. 8.



• = old points
 x = calculated points
 ○ = points used in the next step

fig. 8.

A is a constant.

Depending on the value of f_r four different cases are possible:

(1) If $f_r < f_{\min}$ the search continues in the same direction and points $P_e = C \cdot P_r + (1-C) \cdot \bar{P}$ with function values f_e are calculated for different C values as long as smaller values are encountered. The point with the smallest function value replaces $P_{\max}$. If a large value of C is found all the old points may be changed to points closer to the new minimum value.



fig. 9.

(2) If $f_r \geq f_{\min}$ and at least one point except $P_{\max}$ has a greater function value, $P_{\max}$ is replaced by P_r .
 (3) If $f_r < f_{\max}$ and no point except $P_{\max}$ has a greater function value, $P_{\max}$ is replaced by P_r and the routine continues as in the next case.
 (4) If $f_r \geq f_{\max}$, $P_{c1} = B \cdot P_{\max} + (1-B) \cdot \bar{P}$ is calculated, if the case (3) above occurred $P_{c2} = B \cdot P_r + (1-B) \cdot \bar{P}$ is calculated.



fig. 10.

If the corresponding function value is smaller than $f_{\max}$, the new point replaces $P_{\max}$. Otherwise all points P_i are replaced by $(P_i + P_{\min})/2$.

The iterations continue until $f_{\max} - f_{\min} < EPS$, where EPS is a constant, or until a certain number of iterations (KMAX) has been exceeded.

The subroutine Stepit

The routine and the documentation has been written by Chandler (1965).

The function and the variables are the same as those of the Simplex method just described. The method starts by changing one variable, say v_1 , until a minimum of the function f is found, then changes another variable, say x_1 , until a new minimum is found, and so on. When all variable have been changed twice, all variables are changed at the same time. The changes this time are in proportion to the step sizes earlier used. When a minimum is found, the method starts from the beginning again changing one variable each time, until the step sizes are sufficiently small or a certain number of function calculations has been exceeded.

References

1. Chandler, J. P.: Subroutine Stepit, Quantum Chemistry Program Exchange (QCPE), Dept. of Chemistry, Indiana University, Bloomington, Indiana, USA, 1965.
2. Ekenberg, B.: Spektralanalys med hjälp av bildskärm. (In Swedish.) Rapport LU/CS 75-3. Institutionen för informationsbehandling, särskilt numerisk analys, Sölvegatan 14A, 223 62 Lund, 1975.
3. Nelder, J. A., Mead, R.: A Simplex Method for Function Minimization. Computer Journal 7, 1965. Pages 308-313.
4. Parkinson, J. M., Hutchinson, D.: An Investigation into the Efficiency of Variants on the Simplex Method. From Lootsma, F. A. (ed): Numerical Methods for Non-linear Optimization. Academic Press, London and New York, 1971. Pages 115-135.
5. Ruhe, A., Wedin, P-Å: Algorithms for separable nonlinear least squares problem. Report UMINF-47.74, Umeå University, Institute of Mathematics and Statistics, Dept. of Information processing, 1974.
6. Wedin, P-Å: The non-linear least squares problem from a numerical point of view. Lund University, Department of Computer Sciences, 1972.
7. Wedin, P-Å: On the Gauss-Newton method for the non-linear least squares problem. ITM Arbetsrapport nr 24. Lunds Datacentral, Sölvegatan 18, 223 62 Lund, 1974.
8. Wedin, P-Å, Carlsson, L. Å.: Gaussa. LDC Program Library, Program Abstract, Lund University Computing Centre, Sweden, 1972.

